

In the Claims:

1. (Original) A method of managing a memory that includes a plurality of pages, the method comprising the steps of:

- (a) for each page: identifying a respective risk zone;
- (b) selecting at least one unwritten page for writing new data; and
- (c) writing said new data to said at least one unwritten page only if, for each said at least one unwritten page, said risk zone of said each at least one unwritten page lacks written pages.

2. (Original) The method of claim 1, wherein, if said new data span more than one page, said writing is effected simultaneously.

3. (Original) The method of claim 1, further comprising the step of:

- (d) if said risk zone of one of said at least one unwritten page includes a written page:
 - (i) substituting a different unwritten page for said one of said at least one unwritten page; and
 - (ii) writing at least a portion of said new data to said different unwritten page only if said risk zone of said different unwritten page lacks written pages.

4. (Original) The method of claim 3, wherein step (d) is repeated until all of said new data are written.

5. (Original) The method of claim 1, further comprising the step of:
- (d) for each said selected at least one unwritten page: if said risk zone of said each selected at least one unwritten page includes at least one written page other than said selected at least one unwritten page: marking said each selected unwritten page as deleted.

6. (Original) The method of claim 1, further comprising the steps of: if, for at least one of said at least one unwritten page, said risk zone of said at least one of said at least one unwritten page includes at least one written page:

- (d) selecting a plurality of unwritten pages, including sufficient pages to accommodate both said new data and data stored in said at least one written page;
- (e) copying said data, that are stored in said at least one written page, to said plurality of unwritten pages; and
- (f) writing said new data to said plurality of unwritten pages.

7. (Original) The method of claim 6, wherein said copying of said data that are stored in said at least one written page and said writing of said new data to said plurality of unwritten pages are effected only if, for each page of said plurality of unwritten pages, every page in said risk zone of said each page of said plurality of unwritten pages lacks written pages.

8. (Original) The method of claim 6, further comprising the step of:
- (g) marking said at least one written page as deleted.

9. (Original) The method of claim 6, wherein said copying of said data that are stored in said at least one written page and said writing of said new data to said plurality of unwritten pages are effected simultaneously.

10. (Original) The method of claim 1, wherein all said risk zones include an equal number of pages.

11. (Original) The method of claim 10, wherein each said risk zone includes one page.

12. (Original) The method of claim 1, wherein at least two said risk zones include different numbers of pages.

13. (Original) The method of claim 1, wherein, for each page, said each page and every page of said risk zone of said each page are contiguous.

14. (Original) The method of claim 1, wherein, for at least one page, if said risk zone of said at least one page includes at least one other page, then said at least one page and said risk zone thereof constitute a discontinuous set of pages.

15. (Original) The method of claim 1, wherein, if said new data span more than one page, said selected unwritten pages are written sequentially.

16. (Original) The method of claim 1, wherein, if said new data span more than one page, said selected unwritten pages are written in a random order.

17. (Currently Amended) A data storage system, comprising:

- (a) a data storage medium including a plurality of pages; and
- (b) a controller for writing new data to said plurality of pages in a manner that precludes corruption of old data stored in a first one of said plurality of pages if writing said new data to a second one, of said plurality of pages, that shares at least one cell of said data storage medium with said first one of said plurality of pages, is interrupted, said writing being effected by steps including:
 - (i) identifying a respective risk zone of for each said page of said plurality of pages;
 - (ii) selecting at least one unwritten page of said plurality of pages for writing said new data; and
 - (iii) writing said new data to said at least one unwritten page only if, for each said at least one unwritten page, said risk zone of said each at least one unwritten page lacks written pages.

18. (Original) The system of claim 17, wherein said data storage medium and said controller are operationally associated within a common housing.

19-20. (Canceled)

21. (Original) The system of claim 17, wherein said data storage medium is non-volatile.

22. (Original) The system of claim 21, wherein said data storage medium is a flash memory.

23. (Original) The system of claim 22, wherein said flash memory includes a plurality of multi-level cells.

24. (Original) The system of claim 22, wherein said flash memory includes a plurality of blocks, such that each block of said plurality of blocks includes a respective subplurality of said plurality of pages, and wherein said controller writes said new data to said pages of each said block only sequentially.

25. (Original) The system of claim 22, wherein said flash memory includes a plurality of blocks, such that each block of said plurality of blocks includes a respective subplurality of said plurality of pages, and wherein said controller writes said new data to said pages of each said block in a random order.

26. (Currently Amended) A computer readable storage medium having computer readable code embodied in said computer readable storage medium, the computer readable code for writing new data to at least one of a plurality of pages of a data storage medium in a manner that precludes corruption of old data stored in a first said page if writing said new data to a second said page, that shares at least one cell of the data storage medium with said first page, is interrupted, the computer readable code comprising:

(a) program code for identifying a respective risk zone for each page;

(b) program code for selecting at least one unwritten page for writing new data; and

(c) program code for writing said new data to said at least one unwritten page only if, for each said at least one unwritten page, said risk zone of said each at least one unwritten page lacks written pages.

27. (Canceled)

28. (Currently Amended) A data storage system, comprising:

(a) a data storage medium including a plurality of physical pages; and

(b) a controller for:

(i) accessing said data storage medium as a plurality of logical pages, with each said physical page spanning at least a portion of each of at least two said logical pages of said plurality of logical pages, and

(ii) writing new data to said plurality of logical pages in a manner that precludes corruption of old data stored in a first one of said plurality of logical pages if writing said new data to a second one of said plurality of logical pages, that is spanned by at least one of said physical pages that also spans said first one of said plurality of logical pages, is interrupted, said writing being effected by steps including:

(A) identifying a respective risk zone of each logical page of said plurality of logical pages;

- (B) selecting at least one unwritten logical page of said plurality of logical pages for writing said new data; and
- (C) writing said new data to said at least one unwritten logical page only if, for each said at least one unwritten logical page, said risk zone of said at least one unwritten logical page lacks written logical pages.

29. (Canceled)

30. (Currently Amended) A computer readable storage medium having computer readable code embodied in said computer readable storage medium, the computer readable code comprising:

- (a) program code for accessing, as a plurality of logical pages, a data storage medium that includes a plurality of physical pages, with each said physical page spanning at least a portion of each of at least two said logical pages; and
- (b) program code for writing new data to said plurality of logical pages in a manner that precludes corruption of old data stored in a first one of said plurality of logical pages if writing said new data to a second one of said plurality of logical pages, that is spanned by at least one of said physical pages that also spans said first one of said plurality of logical pages, is interrupted, said program code for writing new data to said plurality of logical pages including:
 - (i) program code for identifying a respective risk zone of each logical page of said plurality of logical pages;

- (ii) program code for selecting at least one unwritten logical page of said plurality of logical pages for writing said new data; and
- (iii) program code for writing said new data to said at least one unwritten logical page only if, for each said at least one unwritten logical page, said risk zone of said at least one unwritten logical page lacks written logical pages.

31. (Canceled)

32. (Original) A data storage system, comprising:

- (a) a data storage medium including a plurality of pages; and
- (b) a controller for writing new data to said plurality of pages by steps including:
 - (i) identifying a respective risk zone of each page of said plurality of pages;
 - (ii) selecting at least one unwritten page of said plurality of pages for writing said new data; and
 - (iii) writing said new data to said at least one unwritten page only if, for each said at least one unwritten page, said risk zone of said each at least one unwritten page lacks written pages.

33. (Original) A computer readable storage medium having computer readable code embodied on ~~an~~ in said computer readable storage medium, the computer readable code for writing new data to at least one of a plurality of pages of a data storage medium, the computer readable code comprising:

- (a) program code for identifying a respective risk zone for each page;
- (b) program code for selecting at least one unwritten page for writing new data; and
- (c) program code for writing said new data to said at least one unwritten page only if, for each said at least one unwritten page, said risk zone of said each at least one unwritten page lacks written pages.